

WHAT IS CLAIMED IS:

1. A system for controlling user interface features of a web application, the system comprising:
  - 5 a collection of user interface control elements, each control element comprising:
    - a namespace;
    - common attributes for defining graphical features of the control element
    - and for associating the control element with the internal state of the core control element;
    - other attributes for defining attributes that affect the intrinsic behavior of
    - 10 the control; and
      - a skin template reference attribute for referencing a skin template;
      - a collection of skin templates comprising extensible markup language based markup contained as children of a container element; and
      - a collection of control element instructions for performing actions associated with
      - 15 the control elements, each script associated with a control element.
  2. The system as claimed in claim 1, further comprising an initialization function for directing the processing one or more control elements in a document object model, the initialization function comprising instructions for traversing each node in a document
  - 20 object model and for searching and calling functions associated with control elements having names following the predetermined naming convention..
  3. The system as claimed in claim 1, wherein the namespace follows a predetermined naming convention comprises having a constant prefix to the name of the element.
  - 25 4. The system as claimed in claim 1, wherein the skin template reference attribute comprises a reference to the location of a skin template file.
  5. The system as claimed in claim 1, wherein the control element is associated with an
  - 30 extensible markup language based element.

6. The system as claimed in claim 5, wherein the control element is a parent of an extensible markup language based element.
7. The system as claimed in claim 5, wherein the control element is a child of an  
5 extensible markup language based element.
8. The system as claimed in claim 2, further comprising:  
a collection of control attributes for adding to existing regular extensible markup  
language based elements in a document object model, the control attributes following the  
10 predetermined naming convention; and  
a collection of control attribute instructions for performing actions associated with  
the collection of control attributes, each instruction associated with a control attribute.
9. The system as claimed in claim 8, wherein the initialization function contains  
15 instructions for traversing each node in the document object model and for searching and  
calling functions associated with control elements and control attributes having names  
following the predetermined naming convention.
10. The system as claimed in claim 1, wherein the collection of control elements  
20 comprises a markup language.
11. The system as claimed in claim 1, wherein the common attributes comprise state  
attributes for specifying the identification of a <state> child element of the control  
element.  
25
12. The system as claimed in claim 1, wherein the common attributes comprise one or  
more of:  
an identification attribute for referencing the control element;  
a label attribute for associating text control;

an x attribute for specifying the x-coordinate of the left edge of the control element;

a y attribute for specifying the y-coordinate of the top edge of the control element;

a width attribute for specifies the width of the control element;

5 a height attribute for specifies the height of the control element;

a preserve aspect ratio attribute for preserving the aspect ratio of the control element when either the width attribute or height attribute is known;

a labelX attribute for specifying the x-coordinate of the left edge of the label, relative to the 'y' attribute;

10 a labelY attribute for specifying the y-coordinate of the bottom edge of the label, relative to the 'x' attribute;

a disabled attribute for specifying whether the control element is disabled and cannot be used;

a state hover attribute for specifying the identification of a <state> child element of the control element, the state hover attribute used to override the appearance of a hover state as defined in a skin of the control element;

15 a state focus attribute for specifying the identification of a <state> child element of the control element, the state focus attribute used to override the appearance of a focus state as defined in a skin of the control element;

20 a state up attribute for specifying the identification of a <state> child element of the control element, the state up attribute used to override the appearance of an up state as defined in a skin of the control element;

a state down attribute for specifying the identification of a <state> child element of the control element, the state down attribute used to override the appearance of a down state as defined in a skin of the control element;

25 a state hit attribute for specifying the identification of a <state> child element of the control element, the state hit attribute used to override the appearance of a hit state as defined in a skin of the control element;

a state disabled up attribute for specifying the identification of a <state> child element of the control element, the state disabled up attribute used to override the appearance of a disabled up state as defined in the skin of the control element; and

5 a state disabled down attribute for specifying the identification of a <state> child element of the control element, the state disabled down attribute used to override the appearance of a disabled down state as defined in a skin of the control element.

13. The system as claimed in claim 12, wherein the set of control elements comprises one or more of:

10 a dsvg:button control element for defining a control that is clicked to trigger an action, the dsvg:button control element comprising:

a namespace following the predetermined naming convention;  
the common attributes;  
other attributes comprising:

15 a toggle attribute for specifying whether the button is a toggle or a sticky button;

a group attribute for specifying the name of a group to which the button control element belongs; and

20 a checked attribute for specifying whether the button control element is down/checked or up/unchecked;

a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

25 a customizable skin template comprising scalable vector graphics markup contained as children of a container element;

a dsvg:comboBox control element for defining a control that is clicked to trigger an action, the dsvg:comboBox control element comprising:

a namespace following the predetermined naming convention;  
the common attributes;

30 other attributes comprising:

a dropdown attribute for specifying whether the comboBox control element has a dropdown list;

an editable attribute for specifying whether the comboBox control element is editable;

5 a value attribute for specifying the value of the label attribute of currently selected item;

a name attribute for specifying the value of a name attribute of a currently selected item; and

a selected identification attribute for specifying the value of the

10 identification attribute of a currently selected item;

a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup

15 contained as children of a container element;

a dsvg:listBox control element for defining a control that is clicked to trigger an action, the dsvg:listBox control element comprising:

a namespace following the predetermined naming convention;

the common attributes;

20 other attributes comprising:

a multi select attribute for specifying whether more than one item can be selected;

an editable attribute for specifying whether the listBox control element is editable;

25 a value attribute for specifying the value of the label attribute of currently selected item;

a name attribute for specifying the value of a name attribute of a currently selected item; and

a selected identification attribute for specifying the value of the

30 identification attribute of a currently selected item;

a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup contained as children of a container element;

5 a dsvg:ListView control element for defining a control that is clicked to trigger an action, the dsvg:ListView control element comprising:

a namespace following the predetermined naming convention;

the common attributes;

10 other attributes comprising:

a multi select attribute for specifying whether more than one item can be selected;

an editable attribute for specifying whether the comboBox control element is editable;

15 a display attribute for specifying a semicolon-delimited list of the names of the attributes in all of the <item> children of a control element;

an anything attribute for specifying an attribute name in which to store data; and

a selected identification attribute for specifying the value of the

20 identification attribute of a currently selected item;

a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup contained as children of a container element;

25 a dsvg:ContextMenu control element for defining a control that is clicked to trigger an action, the dsvg:ContextMenu control element comprising:

a namespace following the predetermined naming convention;

the common attributes;

30 other attributes comprising:

an event source attribute for specifying the identification of an  
 element that triggered the contextMenu control element to appear;  
 a value attribute for specifying the value of the label attribute of  
 currently selected item;  
 5 a name attribute for specifying the value of a name attribute of a  
 currently selected item; and  
 a selected identification attribute for specifying the value of the  
 identification attribute of a currently selected item;  
 a skin template reference attribute for specifying the location of a control  
 10 element skin template, the skin template reference settable to a uniform resource  
 index; and  
 a customizable skin template comprising scalable vector graphics markup  
 contained as children of a container element;  
 a dsvg:item control element for defining a control that is clicked to trigger an  
 15 action, the dsvg:item control element comprising:  
 a namespace following the predetermined naming convention;  
 the common attributes;  
 other attributes comprising:  
 an access key attribute for specifying a shortcut key which, when  
 20 pressed, selects this item; and  
 an anything attribute for specifying an attribute name in which to  
 store data;  
 a skin template reference attribute for specifying the location of a control  
 element skin template, the skin template reference settable to a uniform resource  
 25 index; and  
 a customizable skin template comprising scalable vector graphics markup  
 contained as children of a container element;  
 a dsvg:textbox control element for defining a control that is clicked to trigger an  
 action, the dsvg:textbox control element comprising:  
 30 a namespace following the predetermined naming convention;

the common attributes;

other attributes comprising:

a value attribute for specifies default text within the textbox control element;

5 a num lines attribute for specifying a number of lines allowed in the textbox control element;

a max length attribute for specifying a maximum number of characters allowed in the textbox control element;

a wrap attribute for specifying whether to auto-wrap text;

10 a read only attribute for specifying if the textbox control element is non-editable;

a secret attribute for specifies whether text is secret;

a data type attribute for specifying a type of data that allowed to be entered;

15 a mask attribute for specifies a pattern that allows extra characters to be inserted into data as it is entered and for only allowing specific characters in specific locations; and

a case attribute for specifying the case of data entered into the textbox control element;

20 a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup contained as children of a container element;

25 a dsvg:slider control element for defining a control that is clicked to trigger an action, the dsvg:slider control element comprising:

a namespace following the predetermined naming convention;

the common attributes;

other attributes comprising:

a min attribute for specifying a minimum value of the slider control element;

a max attribute for specifying a maximum value of the slider control element;

5 a min position attribute for specifies a minimum allowed value of a thumb;

a max position attribute for specifies a maximum allowed value of a thumb;

10 a value attribute for specifying an initial value of the slider control element;

an increment attribute for specifying allowed values that the slider control element can create;

a page increment attribute for specifying an amount that a thumb moves;

15 an orientation attribute for specifying a rotation angle of the slider control attribute;

a ticks major attribute for specifying an interval at which major tick marks are displayed; and

a ticks minor attribute for specifying an interval at which minor tick marks are displayed;

20 a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup contained as children of a container element;

25 a dsvg:scrollbar control element for defining a control that is clicked to trigger an action, the dsvg:scrollbar control element comprising:

a namespace following the predetermined naming convention;

the common attributes;

other attributes comprising a bars attribute for specifies the appearance of a horizontal scrollbar or a vertical scrollbar;

a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup contained as children of a container element;

a dsvg:spin control element for defining a control that is clicked to trigger an action, the dsvg:spin control element comprising:

a namespace following the predetermined naming convention;

the common attributes;

other attributes comprising:

a min attribute for specifying a minimum value of the spin control element;

a max attribute for specifying a maximum value of the spin control element;

a value attribute for specifying an initial value of the spin control element; and

an increment attribute for specifying allowed values that the spin control element can create;

a skin template reference attribute for specifying the location of a control element skin template, the skin template reference settable to a uniform resource index; and

a customizable skin template comprising scalable vector graphics markup contained as children of a container element.

14. A system for controlling user interface features of a web application, the system comprising:

a collection of control element instructions for performing actions associated with the control elements, each instruction associated with a control element; and

an initialization function for directing the processing of one or more control elements in a document object model.

15. The system as claimed in claim 14, further comprising a collection of skin templates  
5 comprising extensible markup language based markup contained as children of a container element.

16. A method of controlling user interface features of a web application, the method comprising the steps of:

10        searching for a designated user interface control element in a document object model; and  
          calling a script associated with the designated control element.

17. The method as claimed in claim 16, wherein the step of searching includes the steps  
15 of:

          traversing each node in the document object model; and  
          determining whether an element has a name which follows a designated naming convention.

20 18. The method as claimed in claim 16, wherein the step of calling a script includes the steps of:

          dynamically generating a function name associated with the designated element;  
          passing an object associated with the designated element as a parameter of the generated function;  
25        retrieving the attributes of the object; and  
          performing a function stored in memory having the generated function name.

19. The method as claimed in claim 18, wherein the step of dynamically generating includes the steps of:

30        determining if the name of the designated element contains a designated prefix;

generating a function name comprising of the name of the designated element;  
assigning an object associated with the designated element as the parameter of the  
function; and

5 assigning predetermined instructions of the designated element as steps for the  
function to perform.

20. The method as claimed in claim 16, further comprising the steps of:

searching for a designated attribute in an element in a document object model; and  
calling a script associated with the designated attribute.

10

21. The method as claimed in claim 20, wherein the step of searching for a designated  
attribute comprises the steps of:

searching attributes of an element in a document object model;  
determining whether an element attribute has a name which follows a designated

15 naming convention.

22. The method as claimed in claim 21, wherein the step of calling a script includes the  
steps of:

determining if the name of the designated attribute contains a designated prefix;  
20 generating a function name comprising of the name of the designated attribute;  
assigning an object associated with the designated attribute as the parameter of the  
function name ; and

assigning predetermined instructions of the designated attribute as steps for a  
function having the function name to perform.

25

23. The method as claimed in claim 20, wherein the step of calling a script includes the  
steps of:

dynamically generating a function name associated with the designated attribute;  
passing an object associated with the designated attribute as a parameter of the

30 generated function name;

receiving the attributes of the object; and  
performing a function stored in memory having the generated function name.

24. The method as claimed in claim 23, wherein the step of dynamically generating  
5 comprises the steps of:

determining if the name of the designated attribute contains a designated prefix;  
generating a function name comprising of the name of the designated attribute;  
assigning an object associated with the designated attribute as the parameter of the  
function; and

10 assigning predetermined instructions of the designated attribute as steps for the  
function to perform.

25. A method of controlling user interface features of a web application, the method  
comprising the steps of:

15 adding a behavior element as a child of a user interface control element;  
receiving an event which is equal to an event attribute setting in the behavior  
element; and  
calling a script associated with the behavior element.

20 26. A method of creating a customizable user interface control element having expected  
behaviours, the method comprising the steps of:

categorising user interface controls into fundamental core controls;

determining variations of a core controls;

determining common attributes of the core control;

25 determining fundamental states for the core control;

determining how to allow for absolute positioning of objects the core control;

determining how to allow for absolute customization of appearance of the core

control;

assigning a reference link to the core control;

determining templates for skins to allow for the absolute customization of appearance of the core control;

determining how to associate behaviors to the core control; and

creating a core control element.

5